AMENDMENTS

Amendments to the Specification:

Please replace paragraph 002 with the following amended paragraph:

This application is also a Continuation-in-Part of U.S. Patent Application ~~09/__________,~~ 09/810,148, entitled "System and Method for Discovering Information Objects and Information Object Repositories in Computer Networks", filed March 15, 2001.

Please replace paragraph 025 with the following amended paragraph:

The Cache Resolver is another recent approach to hierarchical Web caching (D. Karger, E. Lehman, T. Leighton, M. Levine, D. Lewin, and R. Panigrahy, "Consistent Hashing and Random Trees: Distributed Caching Protocols for Relieving Hot Spots on the World Wide Web," Proc. 29th ACM Symposium on Theory of Computing (STOC 97), El Paso, Texas, 1997; D. Karger, Sherman, A. Berkheimer, B. Bogstad, R. Dhanidina, K. Iwamoto, B. Kim, L. Matkins, and Y. Yerushalmi, "Web Caching with Consistent Hashing," Proc. 8th International World Wide Web Conference, Toronto, Canada, May 1999). This approach combines hierarchical Web caching with hashing and consists of two main tools, random cache trees and consistent hashing. A tree of Web caches is defined for each information object. When a browser (client) requires an information object, it picks a leaf of the tree and submits a request containing its identifier, the identifier of the object, and the sequence of caches through which the request is to be routed if needed. A Web cache receiving a request ~~it~~ determines if it has a local copy of the page and responds to the request if it does; otherwise, it forwards the request to the next Web cache in the path included in the request. A Web cache starts maintaining a local copy of an information object when the number of requests it receives

for the object reaches a predefined number. A client selects a Web cache by means of consistent hashing, which disseminates requests to leaves of the Web caching hierarchy evenly but, unlike traditional hashing techniques, need not redistribute an updated hash table every time a change occurs in the caching hierarchy (e.g., a new Web cache joins or a Web cache fails). Because caching is difficult to implement or add to existing Web browsers, the Cache Resolver approach implements the hashing in DNS (Internet Domain Name Service) servers modified to fit this purpose. The remaining limitations with this approach stem from the continuing use of a hierarchy of Web caches and the need to implement a hashing function in either Web clients or DNS servers. Routing a request through multiple Web Caches can incur substantial delays for clients to retrieve information objects that are not popular among other clients assigned to the same Web cache by the hashing function. Additional delays, even if small, are incurred at the DNS server that has to provide the address of the Web cache that the client should access. Furthermore, the DNS servers supporting the consistent hashing function must receive information about the loading of all the Web caches in the entire system, or at least a region of the system, in order to make accurate load-balancing decisions.

Please replace paragraph 029 with the following amended paragraph:

Another alternative approach to hierarchical web caching and hash routing protocols consists of forwarding client requests for URLs using routing tables that are very similar to the routing tables used today for the routing of IP packets in the Internet (L. Zhang, S. Michel, S. Floyd, and V. Jacobson, "Adaptive Web Caching: Towards a New Global Caching Architecture," Proc. Third International WWW Caching Workshop, Manchester, England, June 1998, B.S. Michel, K. Nikoloudakis, P. Reiher, and L. Zhang, "URL Forwarding and Compression in Adaptive Web Caching," Proc. IEEE Infocom 2000, Tel Aviv, Israel, April 2000). According to this approach, which is referred to as

"URL request forwarding" herein, Web caches maintain a "URL request routing table" and use it to decide how to forward URL requests to ~~another~~ other Web caches when requested information objects are not found locally. The keys of the URL request routing tables are URL prefixes, which are associated with one ore more identifiers to the next-hop Web caches or cache groups, and a metric reflecting the average delay to retrieve a request from a matching URL.

Please replace paragraph 053 with the following amended paragraph:

In one embodiment of the present invention, a Web router is contacted according to a scheme for enabling the discovery of the caches and servers storing information objects distributed over computer networks, which can be implemented in hardware and/or software, by a client, a Web server, a Web cache, or another type of server with a request for the address of one or more Web caches that a client should contact to obtain an information object. Further descriptions of these various schemes are presented below. A complete description of a Web router is included in co-pending U.S. Patent Application ~~09/____________,~~ 09/810,148, entitled "System and Method for Discovering Information Objects and Information Object Repositories in Computer Networks", filed March 15, 2001, the complete disclosure of which is incorporated by reference herein.

Please replace paragraph 062 with the following amended paragraph:

To reduce communication and processing overhead in Web routers, a topology of Web routers is defined, such that a given Web router has as its neighbor Web routers a subset of all the Web routers in the system (where the term system refers to all or a portion of the virtual network for Web routers discussed above). A Web router may thus be configured with its set of neighbor Web routers. Such a configuration may be a table of neighbor Web routers which is defined by a network service provider and/or is dynamically updated. In another embodiment of the present invention, a Web router

dynamically selects the set of neighbor Web routers with which it should communicate out of all of the Web routers in the system. A Web router preferably communicates with its neighbor Web routers only and uses the Web Information Locator by Distance (WILD) protocol for this purpose. The WILD protocol is disclosed in co-pending and commonly-owned U.S. Provisional Application No. 60/200,401, filed April 28, 2000, now U.S. Patent Application ~~09/~~__________, 09/810,148, filed March ~~16~~ 15, 2001

Please replace paragraph 063 with the following amended paragraph:

In one embodiment of the present invention, WILD runs on top of the Transmission Control Protocol (TCP) in much the same way as the Border Gateway Protocol (BGP) does. In this embodiment, a TCP connection exists between a Web router and each of its neighbor Web routers. In another embodiment of the present invention, WILD can run on top of the TCP Santa Cruz protocol [C. Parsa and J.J. Garcia-Luna-Aceves, "TCP-Santa Cruz: Improving TCP Performance over Networks with Heterogeneous Transmission Media", Proc. IEEE ICNP 99], which is disclosed in commonly-owned U.S. Provisional Application No. 60/190,332, filed on March 16, 2000, now U.S. Patent Application No. ~~09/_,~~ 09/810,148, filed March 15, 2001. Other embodiments of the present invention may be based on alternative protocols for the provision of reliable transmissions between Web routers.

Please replace paragraph 071 with the following amended paragraph:

In a further embodiment, one of the following four mechanisms, or, a combination of some of the following four mechanisms, is or may be used to communicate the best Web cache or content server, or the set of Web caches (more generally the information object repository(ies)), which should serve a client's request:

(1) direct cache selection;

(2) redirect cache selection;

(3) remote DNS cache selection; and

(4) client DNS cache selection.

These approaches are described in detail in co-pending U.S. Provisional Patent Application No.60/200,404, entitled "System and Method for Using a Mapping Between Client Addresses and Addresses of Caches to Support Content Delivery", filed April 28, 2000, and U.S. Patent Application ~~________,~~ 09/843,789, entitled "System and Method for Using a Mapping Between Client Addresses and Addresses of Caches to Support Content Delivery", filed ~~__________,~~ April 26, 2001, the complete disclosure of which is incorporated herein by reference.

Please replace paragraph 077 with the following amended paragraph:

Figure 4 is a flowchart of a method 400 of configuring an anycast cache server in a network upon the loading of content for a URL onto the anycast cache server according to one embodiment of the present invention. As discussed above, the loading of content for a URL triggers the happening of certain configuration steps. In step 402, the anycast cache server receives a request to register the URL (i.e., the URL that is associated with the requested content) in the network routing layer. In ~~the~~ response to this request, in step 404, the anycast cache server maps the URL to a network layer anycast address. In step 406, a local state is created that is adequate for accepting future anycast requests. In step 408, the anycast cache server advertises a route to the anycast IP address with network layer unicast routing. In decision operation 410, it is determined whether further requests can be received at the anycast cache server. If further requests can be received, the process resets and executes again. On the other hand, if no further requests can be received, the process quits.